



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/690,980	10/21/2003	Kenneth P. Sundermeyer	47583/P041US/10311286	4705

59061 7590 07/27/2006

FULBRIGHT & JAWORSKI, LLP (ADOBE)  
2200 ROSS AVENUE  
SUITE 2800  
DALLAS, TX 75201-2784

EXAMINER

RUTLEDGE, AMELIA L

ART UNIT PAPER NUMBER

2176

DATE MAILED: 07/27/2006

Please find below and/or attached an Office communication concerning this application or proceeding.



### **DETAILED ACTION**

1. This action is responsive to communications: Amendment, filed 04/27/2006; Request for Continued Examination, filed 04/27/2006.
2. Claims 1-26 are pending. Claims 1, 8, 15, and 22 are independent claims.

### ***Continued Examination Under 37 CFR 1.114***

3. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 04/27/2006 has been entered.

### ***Claim Rejections - 35 USC § 103***

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

5. **Claims 1-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Brown et al. (hereinafter "Brown"), U.S. Pub. No. 2004/0177321, published September 2004, in view of Dreamweaver TechNote16416, "How to make an**

**inherited editable region uneditable," (hereinafter "Dreamweaver"), last updated 07/06/2002, p. 1-4.**

**Regarding independent claim 1,** Brown teaches a method of inserting XML restriction tags into the editable, i.e., revised, document (p. 9, par. 107-p. 10, par. 109) including instance level restrictions. Brown teaches a method of creating a bounding file or DTD to restrict editing of a formatted structured document (p. 8, par. 94-p. 9, par. 105), by parsing the input document and generating a bounding file (p. 6, par. 74-78); compare to *parsing said multi-formatted document; identifying a plurality of tags responsive to said parsing; generating a revised document based on said multi-formatted document.*

While Brown teaches that the restriction tags in the bounding document are used to determine non-editable content in either structured documents or a GUI, (p. 5, par. 63), i.e., restrict the functions of a page editor; Brown does not explicitly teach a method where the restriction tags are inserted into a generated revised document and restricting edit functions of a page editor editing the revised document where the restricting is responsive to the page editor reading the restriction tags. However, Dreamweaver teaches the use of nested templates with editable and non-editable regions delimited by tags which are inserted into a revised document (p. 1-3), and which restrict the edit functions of the Dreamweaver page editor; compare to *inserting one or more restriction tags into said revised document delimiting non-editable content defined by one or more of said plurality of tags; and restricting edit functions of a page editor editing said revised document, wherein said restricting is responsive to said page editor reading*

*said one or more restriction tags.* The reference *Macromedia Dreamweaver MX: Training from the Source*, published July 23, 2002, Safari Tech Books Online version, <http://proquest.safaribooksonline.com>, p. 1-39; while not being relied upon for the rejections, is cited for the purpose of providing additional information on Dreamweaver and Dreamweaver templates.

Both Brown and Dreamweaver are directed toward editing structured document files. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the templates with editable regions taught by Dreamweaver with the XML bounding DTD with restriction tags taught by Brown, so that the editing GUI would have functionality to control and restrict editing both of HTML (Dreamweaver) and XML (Brown), and so that the user would have the benefit of a simpler way to edit XML files and the application developer would have control over user actions while editing the files (Brown, p. 2, par. 18).

**Regarding dependent claim 2**, Brown teaches compiling a list of document locations containing editable content defined by the tags (Fig. 14).

**Regarding dependent claim 3**, Brown teaches hiding of restriction tags so that the user of a page editor cannot see the tags during editing (p. 6, par. 72).

**Regarding dependent claims 4 and 5**, Brown teaches setting a restriction flag in the revised document to activate edit restriction, by designating a tag editable or non-editable, which is read by the application (p. 8, par. 91-93).

**Regarding dependent claim 6**, Brown teaches the use of shading to notify the user of a restrictive editing location (p. 7, par. 80).

**Regarding dependent claim 7**, while Brown does not teach stripping out the restriction tags when said page editor writes said edited multi-formatted document for presentation, since brown teaches a bounding file, Dreamweaver teaches stripping out the restriction tags when the document is written for presentation, since Dreamweaver teaches stripping out code from nested template documents (p. 3, "Making the region uneditable, but controlling the visibility of its content while in the nested template), and discloses that by modifying or moving code within an editable region, developers can control whether pieces of code are visible or hidden. Both Brown and Dreamweaver are directed toward editing structured document files. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the templates with editable regions taught by Dreamweaver with the XML bounding DTD with restriction tags taught by Brown, so that the editing GUI would have functionality to control and restrict editing both of HTML (Dreamweaver) and XML (Brown), and so that the user would have the benefit of a simpler way to edit XML files and the application developer would have control over user actions while editing the files (Brown, p. 2, par. 18).

**In regard to independent claim 8**, Brown teaches a method of inserting XML restriction tags into the editable, i.e., revised, document (p. 9, par. 107-p. 10, par. 109) including instance level restrictions. Brown teaches a method of creating a bounding file or DTD to restrict editing of a formatted structured document (p. 8, par. 94-p. 9, par. 105), by parsing the tags of the input document and generating a bounding file (p. 6, par. 74-78); *compare to code for parsing said multi-formatted document, wherein said each of said multiple formats is delimited in said multi-formatted document by one or*

*more descriptive labels; code for analyzing said one or more descriptive labels; code for generating a modified document using content of said multi-formatted document.*

While Brown teaches that the restriction tags in the bounding document are used to determine non-editable content in either structured documents or a GUI, (p. 5, par. 63), i.e., restrict the functions of a page editor; Brown does not explicitly teach a method where the code for prohibiting edit functions of a document editor editing the modified document, where the code is executed responsive to the prohibition labels. However, Dreamweaver teaches the use of nested templates with editable and non-editable regions delimited by tags which are inserted into a revised document (p. 1-3), and which restrict the edit functions of the Dreamweaver page editor; compare to *code for inserting a prohibition label in said modified document around each instance of non-editable content as defined by said one or more descriptive labels; and code for prohibiting edit functions of a document editor editing said modified document, wherein said code for prohibiting is executed responsive to said prohibition labels.* The reference *Macromedia Dreamweaver MX: Training from the Source*, published July 23, 2002, Safari Tech Books Online version, <http://proquest.safaribooksonline.com>, p. 1-39; while not being relied upon for the rejections, is cited for the purpose of providing additional information on Dreamweaver and Dreamweaver templates.

Both Brown and Dreamweaver are directed toward editing structured document files. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the templates with editable regions taught by Dreamweaver with the XML bounding DTD with restriction tags taught by Brown, so that the editing GUI

Art Unit: 2176

would have functionality to control and restrict editing both of HTML (Dreamweaver) and XML (Brown), and so that the user would have the benefit of a simpler way to edit XML files and the application developer would have control over user actions while editing the files (Brown, p. 2, par. 18).

**Regarding dependent claims 9-14**, claims 9-14 reflect substantially similar subject matter as claimed in dependent claims 2-7, being directed to the computer program product used to implement the methods as claimed in dependent claims 2-7, and are rejected along the same rationale.

**Independent claim 15 cites:** *A method to restrict editing of a Web document comprising: parsing said Web document; analyzing a plurality of markup tags within said Web document; generating a substitute Web document with content of said Web document; inserting one or more restriction markup tags in said substitute Web document demarcating non-editable content items defined by one or more of said plurality of markup tags; and restricting edit functions of a Web editor editing said substitute Web document, wherein said restricting is responsive to said one or more restriction markup tags.*

Brown teaches a method of inserting XML restriction tags into the editable, i.e., revised, document (p. 9, par. 107-p. 10, par. 109) including instance level restrictions. Brown teaches a method of creating a bounding file or DTD to restrict editing of a formatted structured document (p. 8, par. 94-p. 9, par. 105), by parsing the tags of the input document and generating a bounding file (p. 6, par. 74-78).



While Brown teaches that the restriction tags in the bounding document are used to determine non-editable content in either structured documents or a GUI, (p. 5, par. 63), i.e., restrict the functions of a page editor; Brown does not explicitly teach a method *restricting edit functions of a Web editor editing said substitute Web document*.

However, Dreamweaver teaches the use of nested templates with editable and non-editable regions delimited by tags which are inserted into a generated document (p. 1-3), and which restrict the edit functions of the Dreamweaver page editor. The reference *Macromedia Dreamweaver MX: Training from the Source*, published July 23, 2002, Safari Tech Books Online version, <http://proquest.safaribooksonline.com>, p. 1-39; while not being relied upon for the rejections, is cited for the purpose of providing additional information on Dreamweaver and Dreamweaver templates.

Both Brown and Dreamweaver are directed toward editing structured document files. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the templates with editable regions taught by Dreamweaver with the XML bounding DTD with restriction tags taught by Brown, so that the editing GUI would have functionality to control and restrict editing both of HTML (Dreamweaver) and XML (Brown), and so that the user would have the benefit of a simpler way to edit XML files and the application developer would have control over user actions while editing the files (Brown, p. 2, par. 18).

**Regarding dependent claims 16-21**, claims 16-21 reflect substantially similar subject matter as claimed in dependent claims 2-7, and are rejected along the same rationale.

**6. Claims 22-24, and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Nakao, U.S. Patent No. 6,061,697, issued May 2000, in view of Brown**

**Independent claim 22 cites:** *A system for preserving design elements of a Web page during content editing, said system comprising: a Web development environment comprising: a parsing engine for analyzing a plurality of Web page markup tags; a list of restriction tags for insertion around said design elements, as defined by one or more of said plurality of Web page markup tags; and a page editor comprising: a plurality of deselectable editing functions, wherein said deselection is responsive to said restriction tags.*

Nakao teaches a SGML document editing apparatus which is implemented with a method of parsing a formatted SGML document (Col. 5, l. 5-22; Col. 7, l. 13-25) and identifying a plurality of editable tags in the document (Col. 9, l. 1-45). Nakao teaches a list of restriction tags for insertion around formatting elements. While Nakao teaches a page editor, Nakao does not explicitly teach a page editor comprising a plurality of deselectable editing functions, where deselection is responsive to restriction tags. However, Brown teaches a web development environment and page editor with deselectable editing functions responsive to the restriction tags (p. 6, par. 72; p. 7, par. 80). Brown teaches a method of inserting XML restriction tags into the editable, i.e., revised, document (p. 9, par. 107-p. 10, par. 109) including instance level restrictions. Both inventions are directed toward restricting editing operations on structured

documents. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Brown to Nakao, so that the user would have the benefit of techniques that enable application developers to have some control over what users can do when editing a structured document file (Brown p. 2, par. 21).

**Regarding dependent claim 23**, Nakao teaches a method of parsing a formatted SGML document (Col. 5, l. 5-22; Col. 7, l. 13-25) and identifying a plurality of editable tags in the document (Col. 9, l. 1-45); compare to: *a Web page is processed by said Web development environment to obtain said restriction tags*.

**Regarding dependent claim 24**, while Nakao does not explicitly teach that a subject matter expert operates said page editor to perform said content editing, Brown teaches that the editing system is designed for both application developers and users, where the users operate the restrictive page editor (p. 2, par. 21). The users taught by brown correspond to subject matter experts and would include that group. Both inventions are directed toward restricting editing operations on structured documents. It would have been obvious to one of ordinary skill in the art at the time of the invention to apply Brown to Nakao, so that the user would have the benefit of techniques that enable application developers to have some control over what users can do when editing a structured document file (Brown p. 2, par. 21).

**Regarding dependent claim 26**, while Nakao does not explicitly teach a restriction switch, Brown teaches a restriction switch for deselecting editing functionality (Figure 14) compare to *a restriction switch, accessible by said Web development*

Art Unit: 2176

*environment, for activating deselectability of said plurality of deselectable editing functions.*

**7. Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Nakao in view of Brown, and further in view of Judson, U.S. Patent No. 5,752,643, issued November 1996.**

**Regarding dependent claim 25,** while Nakao in view of Brown does not explicitly teach a cover object for obscuring the restriction tags from view, Judson teaches hiding tags in HTML comment tags and/or by the creation of a covering tag to obscure another tag from view (Col. 5, l. 16-40), compare to *a cover object for obscuring said restriction tags from view in said page editor*. Nakao, Brown, and Judson are all directed toward the presentation and display of structured web documents. It would have been obvious to one of ordinary skill in the art at the time of the invention to combine Nakao, Brown, and Judson, so that the user would have the benefit of viewing the hidden information instantly, without the added time that would have been required to download the information, and information could be viewed as required by the user (Judson, col. 2, l. 12-28).

### ***Response to Arguments***

**8.** Applicant's arguments with respect to claims 1-21 have been considered but are moot in view of the new ground(s) of rejection. The new grounds of rejection includes the removal of the Nakao reference and the addition of the Dreamweaver reference,

which is being relied upon to teach the newly claimed limitation, *restricting edit functions of a page editor editing said revised document, wherein said restricting is responsive to said page editor reading said one or more restriction tags* (Claim 1).

9. Claims 22-25 have not been amended and therefore the previous rejections remain for reasons of record. Applicant's arguments filed 04/27/2006 have been fully considered but they are not persuasive. Brown teaches a method of inserting XML restriction tags into the editable, i.e., revised, document (p. 9, par. 107-p. 10, par. 109) including instance level restrictions. Nakao teaches a SGML document editing apparatus which is implemented with a method of parsing a formatted SGML document (Col. 5, l. 5-22; Col. 7, l. 13-25) and identifying a plurality of editable tags in the document (Col. 9, l. 1-45). While applicant argues the differences between document elements and tags (Remarks, p. 9), it was well known in the art at the time of the invention that when parsing a document, the tags were used to identify the document elements, which as applicant admits, were tags and anything between the opening and closing tags (Remarks, p. 9, par. 4).

10. In response to applicant's argument that there is no suggestion to combine the references, the examiner recognizes that obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either in the references themselves or in the knowledge generally available to one of ordinary skill in the art. See *In re Fine*, 837 F.2d 1071, 5 USPQ2d 1596 (Fed. Cir. 1988) and *In re Jones*, 958 F.2d 347, 21 USPQ2d 1941 (Fed. Cir. 1992). In this case, the motivation

Art Unit: 2176

comes directly from Brown at p. 2, par. 21. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to apply Brown to Nakao, so that the user would have the benefit of techniques that enable application developers to have some control over what users can do when editing a structured document file (Brown p. 2, par. 21).

11. Applicant's arguments regarding the Judson patent used to reject claim 25 (Remarks, p. 13-14) do not dispute the teaching of Judson but rather only refer back to the purported deficiencies of Nakao and Brown, therefore applicant's arguments have been answered above.

12. The examiner agrees with applicant's arguments that in the previous Office Action, the motivation to combine Nakao, Brown, and Judson was insufficient (Remarks, p. 14), and therefore a new motivation statement has been added to the rejection of claim 25.

### ***Conclusion***


Any inquiry concerning this communication or earlier communications from the examiner should be directed to Amelia Rutledge whose telephone number is 571-272-7508. The examiner can normally be reached on Monday - Friday 9:30 - 6:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Heather Herndon can be reached on 571-272-4136. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2176

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

AR

  
HEATHER R. HERNDON  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100